Density-based Clustering: Letting Density Define the Clusters

Ahmed BADI
ahmedbadi905@gmail.com
linkedin.com/in/badi-ahmed

January 3, 2026

Abstract

Density-based clustering is a powerful family of unsupervised learning methods that defines clusters as regions of high point density separated by regions of low density. Unlike centroid-based algorithms such as k-means, density-based methods can discover clusters of arbitrary shape, automatically detect noise and outliers, and do not require the number of clusters to be specified in advance [1]. This article introduces the core ideas behind density-based clustering, focusing on concepts such as ε-neighborhoods, local density, core points, and density reachability. We then present the most important algorithms in this family: DBSCAN, its extension OPTICS for varying densities, and the hierarchical approach HDBSCAN. For each algorithm, we describe the principles, key parameters, strengths, and limitations, and we highlight practical considerations for choosing and tuning density-based clustering methods in real-world applications.

Keywords: Density-based Clustering, DBSCAN, OPTICS, HDBSCAN, Core Points, Noise, Unsupervised Learning.

1 Introduction

Clustering is often explained as the task of grouping similar points together. Centroid-based methods like k-means do this by pulling points towards prototype centers, implicitly assuming that clusters are roughly spherical in the chosen feature space [1]. However, many real datasets do not look like nice round balls: clusters can be elongated, curved, or arranged in complicated shapes (for example, two interlaced "moons" or spiral arms).

Density-based clustering takes a different perspective. Instead of asking "Which centroid is closest to this point?", it asks "Is this point surrounded by many neighbors, or is it isolated?". Clusters are then defined as contiguous regions where the local density of points is high, separated by regions where the density drops below a threshold. Noise and outliers naturally correspond to isolated points in sparse regions.

This philosophy leads to several attractive properties:

- Ability to recover clusters of arbitrary shape.
- Automatic detection of outliers as points lying in low-density regions.
- No need to specify the number of clusters a priori (for DBSCAN/HDBSCAN).

The canonical example in this family is DBSCAN (Density-Based Spatial Clustering of Applications with Noise), but important extensions such as OPTICS and HDBSCAN improve its handling of varying densities.

2 Core Ideas: Density, Neighborhoods, and Reachability

2.1 ε-Neighborhood and Local Density

Most density-based algorithms estimate local point density using a fixed-radius neighborhood. Given a distance function $d(\cdot, \cdot)$ (often Euclidean) and a radius parameter $\varepsilon > 0$, the ε-neighborhood of a point p is defined as:

$$N_\varepsilon(p) = \{q \mid d(p, q) \leq \varepsilon\}. \tag{1}$$

The local density around p is then approximated by the number of points in $N_\varepsilon(p)$.

A second parameter, typically called `MinPts`, sets a minimum density threshold:

- If $|N_\varepsilon(p)| \geq$ `MinPts`, p is considered to lie in a dense region.
- If $|N_\varepsilon(p)|$ is small, p lies in a sparse area.

2.2 Core, Border, and Noise Points

DBSCAN in particular categorizes points into three types [2]:

- **Core point:** A point p such that $|N_\varepsilon(p)| \geq$ `MinPts`. It has enough neighbors to be considered a dense center.
- **Border point:** A point that does not have enough neighbors to be a core point itself, but lies within ε of a core point.
- **Noise point (outlier):** A point that is neither core nor border; it does not belong to any dense region.

Clusters are built by starting from core points and aggregating all points that are *density-reachable* from them; isolated points remain labeled as noise.

2.3 Density Reachability and Connectivity

Two notions are essential:

Directly density-reachable. A point q is directly density-reachable from a core point p if:

$$q \in N_\varepsilon(p) \quad \text{and} \quad p \text{ is a core point.}$$

Density-reachable. A point q is density-reachable from p if there exists a chain of points

$$p = p_1, p_2, \ldots, p_m = q$$

such that each p_{i+1} is directly density-reachable from p_i.

Intuitively, you can "walk" from p to q by stepping from one dense neighborhood to another without jumping through sparse regions. All points that are mutually density-reachable form a **density-connected component**, which DBSCAN identifies as a cluster.

3 DBSCAN: Density-Based Spatial Clustering

3.1 Algorithm and Parameters

DBSCAN is the most widely used density-based clustering algorithm [2]. It relies on two parameters:

- ε (eps): radius of the neighborhood $N_\varepsilon(p)$.

- `MinPts`: minimum number of neighbors required to consider a point as a core point.

Algorithm 1 DBSCAN (high-level)

Require: dataset X, distance d, parameters ε, `MinPts`

Mark all points as unvisited.
for each point p in X **do**
    if p is unvisited **then**
        Mark p as visited.
        Compute $N_\varepsilon(p)$.
        if $|N_\varepsilon(p)| <$ `MinPts` **then**
            Mark p as **noise** (may later become border).
        else
            Create a new cluster C and add p and all points in $N_\varepsilon(p)$ to C.
            For each point q newly added to C:
            if q is unvisited **then**
                Mark q as visited and compute $N_\varepsilon(q)$.
                if $|N_\varepsilon(q)| \geq$ `MinPts` **then**
                    Add all points in $N_\varepsilon(q)$ to cluster C.
                end if
            end if
            Continue expanding C until no new points can be added.
        end if
    end if
end for

Clusters are thus maximal sets of density-connected points; any point not assigned to a cluster is labeled as noise.

3.2 Choosing ε and `MinPts`

Performance of DBSCAN depends heavily on the choice of ε and `MinPts`. Common heuristics include:

- Set `MinPts` to a small integer (e.g. 4 or a bit larger than the dimensionality).
- Use a **k-distance plot**: for each point, compute the distance to its k-th nearest neighbor (with $k =$ `MinPts`), sort these distances, and plot them. The "elbow" or knee in the curve suggests a good ε.

In low dimensions with reasonably separated clusters, such heuristics work well. In high dimensions, the curse of dimensionality makes distances less informative, and DBSCAN may struggle.

3.3 Advantages and Limitations

Advantages [2], [3]:

- No need to choose the number of clusters; it is inferred from the data.
- Can discover clusters of arbitrary shape (e.g. rings, spirals).
- Naturally identifies noise and outliers as low-density points.

Limitations [2], [3]:

- Sensitive to parameter choice (ε, `MinPts`).
- Difficult to handle datasets with clusters of very different densities using a single global ε.
- Performance and distance-based density estimation degrade in very high dimensions.

4 OPTICS: Handling Varying Densities

One of DBSCAN's main challenges is dealing with clusters of different densities. A single global ε may be too small for sparse clusters and too large for dense ones. **OPTICS** (Ordering Points To Identify the Clustering Structure) was proposed to address this problem [4].

4.1 Core Distance and Reachability Distance

OPTICS uses the same parameters ε (as an upper bound) and `MinPts`, but instead of directly assigning clusters, it produces an *ordering* of the points together with two values for each point:

- **Core distance**: the smallest radius needed to enclose `MinPts` neighbors around a point (or undefined if not a core point).
- **Reachability distance**: for a point q with respect to a core point p, defined roughly as

$$\text{reachability}(q \mid p) = \max\big(\text{core\_dist}(p),\ d(p,q)\big).$$

These distances encode how densely connected a point is to its predecessors in the ordering.

4.2 Reachability Plot and Cluster Extraction

The key output of OPTICS is the **reachability plot**: a sequence where each point is plotted in order along the x-axis and its reachability distance is plotted on the y-axis. Valleys in this plot correspond to dense clusters, while peaks represent transitions between clusters or noise.

Clusters can then be extracted by:

- Setting thresholds on reachability distance.
- Identifying valleys and plateaus corresponding to dense regions.
- Potentially obtaining a hierarchical view of clusters at different density levels.

4.3 Benefits Compared to DBSCAN

Advantages of OPTICS [4]:

- Better handling of clusters with varying densities.
- Provides richer information (reachability plot) that reveals hierarchical density structure.
- Can emulate DBSCAN for different ε values from a single run.

The trade-off is increased conceptual and implementation complexity, although many modern libraries provide ready-to-use implementations.

5 HDBSCAN: Hierarchical Density-based Clustering

HDBSCAN (Hierarchical DBSCAN) is another extension that builds a hierarchy of density-based clusters and then extracts a stable flat clustering from this hierarchy. Instead of relying on a single global ε, HDBSCAN varies the density threshold and tracks how clusters appear and disappear.

The main ideas are:

- Construct a minimum spanning tree (MST) of the data using a density-based distance.
- Build a cluster hierarchy as the density threshold changes.
- Quantify the *stability* of each cluster (how long it persists over density levels).
- Select the most stable clusters to form a final flat clustering.

Compared to DBSCAN, HDBSCAN often:

- Handles varying densities more gracefully.
- Requires fewer parameters (typically just `min_cluster_size` and sometimes `min_samples`).

6 When to Use Density-based Clustering?

Density-based methods are particularly attractive when:

- Clusters have irregular or non-convex shapes (e.g. rings, spirals, manifolds).
- There is significant noise or outliers that should not be forced into any cluster.
- The number of clusters is unknown and difficult to estimate.

Typical application domains include:

- **Spatial and geospatial data**: grouping geographic points (e.g. crime locations, GPS traces) into dense hotspots while treating isolated events as noise.
- **Anomaly detection**: isolating rare patterns that do not belong to any dense cluster (fraud, network intrusions).
- **Image and signal analysis**: detecting contiguous dense regions in feature space.

On the other hand, if clusters are roughly spherical, labels are abundant, or interpretability in terms of centroids is crucial, simpler methods like k-means or hierarchical clustering may be preferable [1].

7 Conclusion

Density-based clustering shifts the focus from distance to density: clusters are defined as dense regions separated by sparse gaps, and outliers emerge naturally as points that do not belong to any dense region. DBSCAN embodies this idea with its core/border/noise categorization and its ability to discover arbitrarily shaped clusters without specifying the number of clusters in advance. OPTICS and HDBSCAN extend these principles to datasets with varying densities, providing richer hierarchical views and more robust cluster extractions.

These methods are not without challenges: parameter selection, high dimensionality, and computational cost must be handled carefully. Nevertheless, when their assumptions match the data and the geometry is suitably low- to medium-dimensional, density-based clustering offers a flexible and often very intuitive way to uncover structure and anomalies in unlabeled data.

References

[1] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.

[2] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 226–231.

[3] D. D. Lab, *Density-based clustering and cluster analysis*, `https://domino.ai/blog/topology-and-density-based-clustering`, 2024.

[4] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: Ordering points to identify the clustering structure," in *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*, 1999, pp. 49–60.